Claims

[c1] 1.An application running on a local client for directly communicating with a remote client comprising:

a high-level application with a user interface that receives a command from a user to communicate with the remote client;

a transport layer, coupled to the high-level application, for generating and receiving Transport-Control-Protocol (TCP) packets and User Datagram Protocol (UDP) packets;

a socket subsystem, receiving TCP and UDP packets from the transport layer, for sending and receiving packets from an external network connected to the remote client and an external manager, the remote client and the external manager separated from the local client by a local firewall that protects the local client from un-requested packets; and

a null-packet generator, coupled to the high-level application, for generating a null UDP packet to a remote UDP port of the remote client in response to a TCP packet from the external manager that contains an address of the remote client with an identifier for the remote UDP port, the transport layer and the socket subsystem sending the null UDP packet from a local UDP port through the local firewall toward the remote client;

wherein the local firewall opens a window between the local UDP port and the remote UDP port in response to the null UDP packet, the socket subsystem and transport layers receiving UDP packets containing user data from the remote client through the window in the local firewall,

whereby the window in the local firewall is opened by the null UDP packet sent in response to the TCP packet from the external manager.

[c2] 2.The application of claim 1 wherein the user data in the UDP packets represents audio sounds, the application further comprising:

a multimedia subsystem, coupled to the high-level application, for receiving voice from the user for transmission to the remote client in the UDP packets, and for playing as audio sounds to the user the user data received from the remote client in the UDP packets.

[c3] 3.The application of claim 2 further comprising:

a registration module, in the high-level application, for instructing the transport layer and socket subsystem to send a registration packet to the external manager, the registration packet being a TCP packet that contains an identifier for the local UDP port,

whereby the registration module registers the local UDP port with the external manager using a TCP packet.

[c4] 4.The application of claim 2 further comprising:

a reply module, in the high-level application, for instructing the transport layer and socket subsystem to send a reply packet to the external manager, the reply packet being a TCP packet that indicates to the external manager that the null UDP packet has been sent to open the window in the local firewall,

whereby the reply module indicates transmission of the null UDP packet using a TCP packet to the external manager.

[c5] 5.The application of claim 4 wherein the null UDP packet has a data payload size of zero, and a message identifier that identifies the null UDP packet as being a null packet, the null UDP packet containing no audio data played to a user,

whereby the null UDP packet opens the window in the local firewall but does not contain audio data.

[c6] 6.The application of claim 5 wherein TCP packets form a TCP connection to the external manager initiated by a SYN packet, a SYN+ACK packet, and an ACK packet,

wherein TCP connections are made to the external manager, but UDP packets flow between the local client and the remote client,

wherein the external network is an Internet.

[c7] 7.The application of claim 2 wherein the window in the local firewall is used for a two-way direct communication channel between the local UDP port of the local client, and the remote UDP port of the remote client, wherein UDP packets containing audio or video data are transmitted in two directions between the remote and local clients through the window in the local firewall.

[c8] 8.A computer-implemented method for directly communicating between a first computer and a second computer using a third computer to establish communication comprising:

sending a call request from the first computer to the third computer, the call request identifying the second computer;

sending a message from the third computer to the second computer in response to the call request, the message identifying a first address of the first computer;

receiving the message at the second computer and generating a firewall-opening packet that is destined to the first address of the first computer;

sending the firewall-opening packet from the second computer toward the first computer;

opening a window in a firewall that protects the second computer from receiving un-requested packets when the firewall-opening packet is sent by the second computer, the window allowing packets from the first computer to reach the second computer through the firewall; and

sending direct communication packets from the first computer to the second computer through the window in the firewall created by the firewall-opening packet sent by the second computer in response to the message from the third computer,

whereby the window in the firewall protecting the second computer is created for use by the first computer.

[c9] 9.The computer-implemented method of claim 8 wherein the firewall-opening packet is a User Datagram Protocol (UDP) packet and wherein the direct communication packets are UDP packets.

[c10] 10.The computer-implemented method of claim 9 wherein the firewall-opening packet is a null packet having no data in a data payload.

[c11] 11.The computer-implemented method of claim 10 wherein the first address of the first computer sent in the message from the third computer to the second computer comprises an IP address and a UDP port of the first computer.

[c12] 12.The computer-implemented method of claim 9 further comprising:

sending a reply message from the second computer to the third computer once the firewall-opening packet has been sent;

sending a start message from the third computer to the first computer in response to the reply message, the start message instructing the first computer to send the direct communication packets directly to the second computer through the window in the firewall.

[c13] 13.The computer-implemented method of claim 12 wherein sending the call request and sending the reply message comprise communicating with port 80 of the third computer using a Transport-Control-Protocol (TCP);

wherein sending the direct communication packets comprises sending a flow of UDP packets from a first UDP port of the first computer to a second UDP port of the second computer,

whereby communication to the third computer uses TCP while direct communication between the first and second computers uses UDP.

[c14] 14.The computer-implemented method of claim 13 wherein the first UDP port is included in the first address of the message sent from the third computer to the second computer;

wherein the second UDP port is included in the start message from the third computer to the first computer;

wherein the firewall-opening packet includes both the first address with the first UDP port and a second address with the second UDP port,

wherein the window in the firewall is a window for UDP packets between the first UDP port of the first computer and the second UDP port of the second computer.

[c15] 15.The computer-implemented method of claim 14 further comprising:

sending a first register message from the first computer to the third computer, the first register message including an indication of the first UDP port;

storing the indication of the first UDP port in association with a registration entry for the first computer in a registration directory accessible by the third computer;

sending a second register message from the second computer to the third computer, the second register message including an indication of the second UDP port; and

storing the indication of the second UDP port in association with a registration entry for the second computer in the registration directory accessible by the third computer,

whereby the first and second computer register with the third computer.

[c16] 16.The computer-implemented method of claim 15 further comprising:

sending direct communication packets from the second UDP port of the second computer to the first UDP port of the first computer,

whereby two-way communication is established directly between the first and second computers with assistance of the third computer.

[c17] 17.The computer-implemented method of claim 16 further comprising:

creating a second window in a second firewall that protects the first computer when a first of the direct communication packets is sent from the first computer to the second computer, the second window allowing direct communication packets from the second computer to pass through the second firewall.

[c18] 18.The computer-implemented method of claim 17 further comprising:

discarding the firewall-opening packet at the second firewall, the second firewall blocking the firewall-opening packet from reaching the first computer, whereby the second firewall blocks the firewall-opening packet as an un-requested packet.

[c19] 19.A computer-program product comprising:

a computer-usable medium having computer-readable program code means embodied therein for directly communicating with a remote peer through a firewall, the computer-readable program code means in the computer-program product comprising:

network connection means for transmitting and receiving packets from an Internet, the packets including Transmission Control Protocol (TCP) packets from an external manager, and User Datagram Protocol (UDP) packets from the remote peer;

firewall-opening packet means, receiving a TCP message from the external manager with a remote UDP port of the remote peer, for generating a firewall-opening packet that is destined to the remote UDP port of the remote peer;

the network connection means also for sending the firewall-opening packet from a local UDP port, the firewall-opening packet destined for the remote UDP port of the remote peer;

wherein a window in the firewall is created when the firewall-opening packet is sent, the window allowing packets from the remote peer to reach the network connection means through the firewall; and

direct communication means, coupled to the network connection means, for sending UDP packets from the local UDP port to the remote UDP port of the remote peer through the window in the firewall created by the firewall-opening packet,

whereby the window in the firewall is created for use by the remote peer.

[c20] 20.The computer-program product of claim 19 wherein the computer-readable program code means in the computer-program product further comprises:

call request means for using TCP packets to send a call request to the external manager, the call request identifying the remote peer, the external manager sending a message identifying the local UDP port to the remote peer in response to the call request.